## AMENDMENTS TO THE CLAIMS

1	1.	(Previously presented) A method of automatically generating a keystream
2		segment of an arbitrary location of a complete keystream of an additive stream cipher,
3		the method comprising the computer-implemented steps of:
4		receiving a location value that identifies a location of the keystream segment within
5		the complete keystream;
6		creating and storing a state value for a leaf node of a balanced binary tree, wherein the
7		leaves of the tree represent the complete keystream and the leaf node
8		represents the keystream segment at the location, by a preorder traversal of the
9		tree from root node to the leaf node wherein a leftward tree branch transition
10		comprises computing a first non-linear function and a rightward tree branch
11		transition comprises computing a second non-linear function;
12		creating and storing the keystream segment by applying a third function to the state
13		value of the leaf node.
1	2.	(Previously presented) A method as recited in Claim 1, further comprising the
2		steps of creating and storing the balanced binary tree by creating and storing a stack of
3		$h$ elements wherein the $i^{th}$ element of said stack stores a state datum for the $i^{th}$ node on
4		a path from a root node of the tree to the leaf node.
1	3.	(Previously presented) A method as recited in Claim 2, wherein the step of
2		creating and storing a state value for a leaf node comprises the steps of computing and
3		storing a state value for the leaf node that is unique with respect to any other state
4		value that is computed at any other time for any other leaf node of the tree.
1	4.	(Canceled.)

- 1 5. (Previously presented) The method as recited in Claim 1, wherein each leaf
  2 node stores m bits of state information, wherein m is a multiple of twelve.
- 1 6. (Currently amended) The method as recited in Claim 1, further comprising
- 2 the steps of:
- 3 creating and storing m=3n bits of state information in each leaf node comprising a
- 4 concatenation of three n bit quantities z|y|x, wherein n is a multiple of four;
- computing the first non-linear function a and the second non-linear function b as the
- 6 composition of a diffusion function d with the nonlinear "confusion" functions
- 7 f and g, wherein a = f od and b = g od and wherein
- 8 f(z | y | x) = 2z | S(R(S(R(y)))) | L(S(L(S(x))))
- 9 g(z | y | x) = 2z+1|L(S(L(S(y))))|S(R(S(R(x))))
- 10 d(z | y | x) = z | x + y + z | 2x + y + z
- $11 c(z \mid y \mid x) = x \oplus y$
- wherein integer addition modulo two is denoted as +, bitwise exclusive-or is denoted
- as  $\oplus$ , and bitwise complementation is denoted as  $\neg$ ;
- wherein the R denotes rotation by n/4 bits to in a direction of a least significant bit
- and L denotes rotation by n/4 bits in a direction of a most significant bit; and
- wherein a nonlinear function S comprises a lookup in a key-dependent substitution
- 17 table.
- 1 7. (Previously presented) The method as recited in Claim 1, wherein the third
- 2 function comprises computing a linear reduction of 2n bits of the state value to n bits
- 3 thereof.
- 1 8. (Previously presented) A method as recited in Claim 6, wherein the third
- function comprises computing a bitwise Boolean exclusive OR of x and y.

A method as recited in Claim 6, further comprising the 9. (Previously presented) 1 2 steps of creating and storing the substitution table S by selecting four invertible functions and storing the four invertible functions in a concatenated form. 3 A method as recited in Claim 6, further comprising the 1 10. (Previously presented) 2 steps of computing functions f and g in seven instructions of a central processing unit 3 that can issue two instructions simultaneously, by using five registers to store values of x, y, z, a temporary variable, and a pointer to the substitution table S. 4 A method as recited in Claim 6, wherein the 1 11. (Previously presented) 2 substitution table S comprises an array of key dependent pseudorandom integer 3 values. 1 12. (Previously presented) A method as recited in Claim 6, wherein the 2 substitution table S comprises an array of 256 key dependent pseudorandom 32-bit 3 unsigned integer values. The method as recited in Claim 1, further comprising 1 13. (Previously presented) 2 the steps of creating and storing a key for use by the first non-linear function and the 3 second non-linear function, wherein the key comprises a table of key dependent 4 pseudorandom values. The method as recited in Claim 1, further comprising 1 14. (Previously presented) 2 the steps of creating and storing, once and at a time prior to receiving the location 3 value, a key for use by the first non-linear function and the second non-linear 4 function, wherein the key comprises a table of key dependent pseudorandom values.

I	15.	(Previously presented) The method as recited in Claim 1, further comprising
2		the steps of creating and storing a key in the form of a plurality of pseudo-randomly
3		selected invertible functions, wherein each of the invertible functions maps an 8-bit
4		portion of the state value to an 8-bit quantity for use as a substitute portion of the state
5		value.
1	16.	(Previously presented) A method as recited in Claim 1, wherein the pseudo-
2		randomly selected invertible functions are stored in a plurality of substitution tables,
3		and wherein the plurality of substitution tables are generated by:
4		setting each of the plurality of substitution tables equal to the identity function;
5		for each element of each of the plurality of substitution tables, swapping said element
6		with another element of such table in a key-dependent manner, and also
7		performing the same swapping operation on each table that has been
8		previously been generated.
1	17.	(Previously presented) A method of enciphering a plaintext using at least one
2		keystream segment at an arbitrary location of a complete keystream, the method
3		comprising the computer-implemented steps of:
4		receiving a segment of a plaintext;
5		receiving a location value that identifies a location of the keystream segment within
6		the complete keystream;
7		creating and storing a state value for a leaf node of a balanced binary tree, wherein the
8		leaves of the tree represent the complete keystream and the leaf node
9		represents the keystream segment at the location, by a preorder traversal of the
10		tree from root node to the leaf node wherein a leftward tree branch transition
11		comprises computing a first non-linear function and a rightward tree branch
12		transition comprises computing a second non-linear function;

13		creating and storing the keystream segment by applying a third function to the state
14		value of the leaf node;
15		enciphering the segment of the plaintext by combining the keystream segment with
16		the segment of the plaintext using a Boolean exclusive OR operation to result
17		in creating and storing a segment of ciphertext.
1	18.	(Previously presented) A method of encrypting an ordered plurality of packets
2		of a network communication link using at least one keystream segment at an arbitrary
3		location of a complete keystream, the method comprising the computer-implemented
4		steps of:
5		receiving a packet from among the plurality of packets;
6		determining a location value that represents a relative location of the packet among
7		the plurality of packets;
8		creating and storing a state value for a leaf node of a balanced binary tree, wherein the
9		leaves of the tree represent the complete keystream and the leaf node
10		represents a keystream segment at the relative location, by a preorder traversal
11		of the tree from root node to the leaf node wherein a leftward tree branch
12		transition comprises computing a first non-linear function and a rightward tree
13		branch transition comprises computing a second non-linear function;
14		creating and storing the keystream segment by applying a third function to the state
15		value of the leaf node;
16		enciphering the packet by combining the keystream segment with data of the packet
17		using a Boolean exclusive OR operation to result in creating and storing
18		enciphered packet data.
1	19.	(Previously presented) A computer-readable medium carrying one or more
2		sequences of instructions for automatically generating a keystream segment of an
3		arbitrary location of a complete keystream of an additive stream cipher, which

4		instructions, when executed by one or more processors, cause the one or more
5		processors to carry out the steps of:
6		receiving a location value that identifies a location of the keystream segment within
7		the complete keystream;
8		creating and storing a state value for a leaf node of a balanced binary tree, wherein the
9		leaves of the tree represent the complete keystream and the leaf node
10		represents the keystream segment at the location, by a preorder traversal of the
11		tree from root node to the leaf node wherein a leftward tree branch transition
12		comprises computing a first non-linear function and a rightward tree branch
13		transition comprises computing a second non-linear function;
14		creating and storing the keystream segment by applying a third function to the state
15		value of the leaf node.
1	20.	(Previously presented) An apparatus for automatically generating a keystream
2		segment of an arbitrary location of a complete keystream of an additive stream cipher,
3		comprising:
4		means for receiving a location value that identifies a location of the keystream
5		segment within the complete keystream;
6		means for creating and storing a state value for a leaf node of a balanced binary tree,
7		wherein the leaves of the tree represent the complete keystream and the leaf
8		node represents the keystream segment at the location, by a preorder traversal
9		of the tree from root node to the leaf node wherein a leftward tree branch
10		transition comprises computing a first non-linear function and a rightward tree
11		branch transition comprises computing a second non-linear function;
12		means for creating and storing the keystream segment by applying a third function to

I	21.	(Amended) An apparatus for automatically generating a keystream segment of an
2		arbitrary location of a complete keystream of an additive stream cipher, comprising:
3		a network interface that is coupled to the data network for receiving one or more
4		packet flows therefrom;
5		a processor;
6		one or more stored sequences of instructions which, when executed by the processor,
7		cause the processor to carry out the steps of:
8		receiving a location value that identifies a location of the keystream segment
9		within the complete keystream;
10		creating and storing a state value for a leaf node of a balanced binary tree,
11		wherein the leaves of the tree represent the complete keystream and the
12		leaf node represents the keystream segment at the location, by a
13		preorder traversal of the tree from root node to the leaf node wherein a
14		leftward tree branch transition comprises computing a first non-linear
15		function and a rightward tree branch transition comprises computing a
16		second non-linear function;
17		creating and storing the keystream segment by applying a third function to the
18		state value of the leaf node.
1	22.	(New) A computer-readable medium as recited in Claim 19, comprising further
2		sequences of instructions which, when executed by the one or more processors, cause
3		the one or more processors to perform the steps of any of Claims 2, 3, 5, 6, 7, 8, 9, 10,
4		11, 12, 13, 14, 15, or 16.
1	23.	(New) An apparatus as recited in Claim 20, further means for performing functions
2		recited in the steps of any of Claims 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, or 16.

Ser. No. 09/675,570 filed Sept. 29, 2000 GAU 2134--Examiner J. Lipman Atty Docket 50325-0512 (Seq. No. 3245)

1	24.	(New) An apparatus as recited in Claim 21, wherein the sequences of instructions
2		comprise further sequences of instructions which, when executed by the processor,
3		cause the processor to perform the steps of any of Claims 2, 3, 5, 6, 7, 8, 9, 10, 11, 12,
4		13, 14, 15, or 16.
5		